

**In the Claims:**

Please amend claims 12, 22, and 32. The claims are as follows:

1-11 (Canceled)

12. (Currently amended) A method for hyperlinking a main file with N target files, said main file and said N target files being stored in a computer readable medium of a computer system, said N at least 1, said main file having a primary filename of a form F.E, said F representing a name component of the primary filename, said E representing an extension component of the primary filename, said N target files denoted as  $T_1, T_2, \dots, T_N$  having an associated N target file addresses respectively denoted as  $A_1, A_2, \dots, A_N$ , said method comprising:

encoding hyperlinking the primary filename with the N target file addresses to form a composite filename having a form of  $F(A).E$ , said A representing  $A_1, A_2, \dots, A_N$ , said symbol ( between said F and said A denoting a first control character that separates said F and said A, said symbol ) between said A and said E denoting a second control character that separates said A and said E, said second control character differing from the first control character; and

storing the composite filename in said computer readable medium.

13. (Previously presented) The method of claim 12, wherein the target file  $T_i$  is a source file of the main file such that  $A_i$  is a source file address of the source file  $T_i$ , and wherein i is selected from the group consisting of 1, 2, ..., and N.

14. (Previously presented) The method of claim 13, wherein the source file address  $A_i$  is a Universal Resource Locator (URL) of an Internet web page.

15. (Previously presented) The method of claim 12, wherein  $N$  is at least 2, wherein  $A$  is represented in a form of  $A_1\{A_2\{ \dots \{A_N$ , and wherein said symbol  $\{$  denotes a third control character that separates  $A_{n-1}$  and  $A_n$  for  $n=2, \dots, N$ , and wherein the third control character differs from both the first control character and the second control character.

16. (Previously presented) The method of claim 15, said method further comprising determining whether corresponding path portions of two consecutive target file addresses  $A_i$  and  $A_{i+1}$  is a common path, wherein  $i$  is selected from the group consisting of 1, 2, ..., and  $N-1$ , and if said determining determines that said corresponding path portions is a common path then:

inserting a fourth control character denoted by a symbol  $\}$  at the end of the common path of  $A_i$ , wherein the fourth control character differs from the first control character, the second control character, and the third control character; and

removing the common path from  $A_{i+1}$ .

17. (Previously presented) The method of claim 12, said method further comprising specifying predetermined character strings and associated substitute characters; and for  $n=1, 2, \dots, N$ :

identifying in  $A_n$  at least one character string of said predetermined character strings; and  
replacing in  $A_n$  each identified character string with its associated substitute character.

18. (Previously presented) The method of claim 17, wherein an identified character string consists of one character.

19. (Previously presented) The method of claim 17, wherein an identified character string consists of at least two characters.

20. (Previously presented) The method of claim 12, said method further comprising:

decoding the composite filename, by parsing the composite filename, to extract from the composite filename a target file address  $A_i$  of the  $N$  target file addresses, wherein  $i$  is selected from the group consisting of 1, 2, ..., and  $N$ ; and

accessing the target file  $T_i$  at the target file address  $A_i$ .

21. (Previously presented) The method of claim 20, said method further comprising:

analyzing the target file address  $A_i$  to determine a file type of the target file  $T_i$ ; and

launching an application that is associated to the file type of the target file  $T_i$ .

22. (Currently amended) An apparatus comprising a computer system, said computer system comprising a computer readable medium, said computer readable medium comprising software adapted to be executed by the computer system to implement a method for hyperlinking a main file with  $N$  target files, said main file and said  $N$  target files being stored in the computer system, said  $N$  at least 1, said main file having a primary filename of a form  $F.E$ , said  $F$  representing a name component of the primary filename, said  $E$  representing an extension component of the

primary filename, said N target files denoted as  $T_1, T_2, \dots, T_N$  having an associated N target file addresses respectively denoted as  $A_1, A_2, \dots, A_N$ , said method comprising:

encoding hyperlinking the primary filename with the N target file addresses to form a composite filename having a form of  $F(A).E$ , said A representing  $A_1, A_2, \dots, A_N$ , said symbol ( between said F and said A denoting a first control character that separates said F and said A, said symbol ) between said A and said E denoting a second control character that separates said A and said E, said second control character differing from the first control character; and  
storing the composite filename in said computer readable medium.

23. (Previously presented) The apparatus of claim 22, wherein the target file  $T_i$  is a source file of the main file such that  $A_i$  is a source file address of the source file  $T_i$ , and wherein i is selected from the group consisting of 1, 2, ..., and N.

24. (Previously presented) The apparatus of claim 23, wherein the source file address  $A_i$  is a Universal Resource Locator (URL) of an Internet web page.

25. (Previously presented) The apparatus of claim 22, wherein N is at least 2, wherein A is represented in a form of  $A_1\{A_2\{ \dots \{A_N$ , and wherein said symbol { denotes a third control character that separates  $A_{n-1}$  and  $A_n$  for  $n=2, \dots, N$ , and wherein the third control character differs from both the first control character and the second control character.

26. (Previously presented) The apparatus of claim 25, said method further comprising determining whether corresponding path portions of two consecutive target file addresses  $A_i$  and  $A_{i+1}$  is a common path, wherein  $i$  is selected from the group consisting of 1, 2, ..., and  $N-1$ , and if said determining determines that said corresponding path portions is a common path then:

inserting a fourth control character denoted by a symbol  $\}$  at the end of the common path of  $A_i$ , wherein the fourth control character differs from the first control character, the second control character, and the third control character; and

removing the common path from  $A_{i+1}$ .

27. (Previously presented) The apparatus of claim 22, said method further comprising specifying predetermined character strings and associated substitute characters; and for  $n=1, 2, \dots, N$ :

identifying in  $A_n$  at least one character string of said predetermined character strings; and

replacing in  $A_n$  each identified character string with its associated substitute character.

28. (Previously presented) The apparatus of claim 27, wherein an identified character string consists of one character.

29. (Previously presented) The apparatus of claim 27, wherein an identified character string consists of at least two characters.

30. (Previously presented) The apparatus of claim 22, said method further comprising:

decoding the composite filename, by parsing the composite filename, to extract from the composite filename a target file address  $A_i$  of the  $N$  target file addresses, wherein  $i$  is selected from the group consisting of 1, 2, ..., and  $N$ ; and

accessing the target file  $T_i$  at the target file address  $A_i$ .

31. (Previously presented) The apparatus of claim 30, said method further comprising:

analyzing the target file address  $A_i$  to determine a file type of the target file  $T_i$ ; and

launching an application that is associated to the file type of the target file  $T_i$ .

32. (Currently amended) A computer readable medium comprising software adapted to be executed by a computer system to implement a method for hyperlinking a main file with  $N$  target files, said main file and said  $N$  target files being stored in the computer system, said  $N$  at least 1, said main file having a primary filename of a form  $F.E$ , said  $F$  representing a name component of the primary filename, said  $E$  representing an extension component of the primary filename, said  $N$  target files denoted as  $T_1, T_2, \dots, T_N$  having an associated  $N$  target file addresses respectively denoted as  $A_1, A_2, \dots, A_N$ , said method comprising:

encoding hyperlinking the primary filename with the  $N$  target file addresses to form a composite filename having a form of  $F(A).E$ , said  $A$  representing  $A_1, A_2, \dots, A_N$ , said symbol ( between said  $F$  and said  $A$  denoting a first control character that separates said  $F$  and said  $A$ , said symbol ) between said  $A$  and said  $E$  denoting a second control character that separates said  $A$  and said  $E$ , said second control character differing from the first control character; and

storing the composite filename in said computer readable medium.

33. (Previously presented) The computer readable medium of claim 32, wherein the target file  $T_i$  is a source file of the main file such that  $A_i$  is a source file address of the source file  $T_i$ , and wherein  $i$  is selected from the group consisting of 1, 2, ..., and  $N$ .

34. (Previously presented) The computer readable medium of claim 33, wherein the source file address  $A_i$  is a Universal Resource Locator (URL) of an Internet web page.

35. (Previously presented) The computer readable medium of claim 32, wherein  $N$  is at least 2, wherein  $A$  is represented in a form of  $A_1\{A_2\{... \{A_N$ , and wherein said symbol  $\{$  denotes a third control character that separates  $A_{n-1}$  and  $A_n$  for  $n=2, ..., N$ , and wherein the third control character differs from both the first control character and the second control character.

36. (Previously presented) The computer readable medium of claim 35, said method further comprising determining whether corresponding path portions of two consecutive target file addresses  $A_i$  and  $A_{i+1}$  is a common path, wherein  $i$  is selected from the group consisting of 1, 2, ..., and  $N-1$ , and if said determining determines that said corresponding path portions is a common path then:

inserting a fourth control character denoted by a symbol  $\}$  at the end of the common path of  $A_i$ , wherein the fourth control character differs from the first control character, the second control character, and the third control character; and

removing the common path from  $A_{i+1}$ .

37. (Previously presented) The computer readable medium of claim 32, said method further comprising specifying predetermined character strings and associated substitute characters; and for  $n=1, 2, \dots, N$ :

identifying in  $A_n$  at least one character string of said predetermined character strings; and

replacing in  $A_n$  each identified character string with its associated substitute character.

38. (Previously presented) The computer readable medium of claim 37, wherein an identified character string consists of one character.

39. (Previously presented) The computer readable medium of claim 37, wherein an identified character string consists of at least two characters.

40. (Previously presented) The computer readable medium of claim 32, said method further comprising:

decoding the composite filename, by parsing the composite filename, to extract from the composite filename a target file address  $A_i$  of the  $N$  target file addresses, wherein  $i$  is selected from the group consisting of 1, 2, ..., and  $N$ ; and

accessing the target file  $T_i$  at the target file address  $A_i$ .

41. (Previously presented) The computer readable medium of claim 40, said method further comprising:



analyzing the target file address  $A_i$  to determine a file type of the target file  $T_i$ ; and  
launching an application that is associated to the file type of the target file  $T_i$ .